

*NASA Contractor Report 198306*

*ICASE Report No. 96-20*

# ICASE

## **PATTERN SEARCH ALGORITHMS FOR BOUND CONSTRAINED MINIMIZATION**

**Robert Michael Lewis  
Virginia Torczon**

*NASA Contract No. NAS1-19480  
March 1996*

*Institute for Computer Applications in Science and Engineering  
NASA Langley Research Center  
Hampton, VA 23681-0001*

*Operated by Universities Space Research Association*



*National Aeronautics and  
Space Administration*

*Langley Research Center  
Hampton, Virginia 23681-0001*

19960514 075

DTIC QUALITY INSPECTED 1

**DISTRIBUTION STATEMENT A**

Approved for public release;  
Distribution Unlimited

# Pattern Search Algorithms for Bound Constrained Minimization<sup>1</sup>

Robert Michael Lewis  
Institute for Computer Applications in Science and Engineering  
Mail Stop 132C, NASA Langley Research Center  
Hampton, Virginia 23681-0001

Virginia Torczon<sup>2</sup>  
Department of Computer Science  
College of William and Mary  
Williamsburg, Virginia 23187-8795

## Abstract

We present a convergence theory for pattern search methods for solving bound constrained nonlinear programs. The analysis relies on the abstract structure of pattern search methods and an understanding of how the pattern interacts with the bound constraints. This analysis makes it possible to develop pattern search methods for bound constrained problems while only slightly restricting the flexibility present in pattern search methods for unconstrained problems. We prove global convergence despite the fact that pattern search methods do not have explicit information concerning the gradient and its projection onto the feasible region and consequently are unable to enforce explicitly a notion of sufficient feasible decrease.

---

<sup>1</sup> This research was supported in part by the National Aeronautics and Space Administration under NASA Contract No. NAS1-19480 while the authors were in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA 23681-0001.

<sup>2</sup> This research was supported by the Air Force Office of Scientific Research grant F49620-95-1-0210, Sandia National Laboratories, Lawrence Livermore, RFQ LC-2683, and the Center for Research on Parallel Computation through NSF Cooperative Agreement 9120008.

**1. Introduction.** This paper extends the class of pattern search methods for unconstrained minimization, considered in [14], to bound constrained problems:

$$(1) \quad \begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & \ell \leq x \leq u, \end{array}$$

where  $f : \mathbf{R}^n \rightarrow \mathbf{R}$ ,  $\ell, x, u \in \mathbf{R}^n$ , and  $\ell < u$ . We allow the possibility that some of the variables are unbounded either above or below by permitting  $\ell_j, u_j = \pm\infty$ ,  $j = 1, \dots, n$ .

Our convergence analysis is guided by that for pattern search methods for unconstrained problems [14]. We can guarantee that if the objective  $f$  is continuously differentiable, then a subsequence of the iterates produced by a pattern search method for problems with bound constraints converges to a stationary point of problem (1). By a stationary point of problem (1) we mean a feasible point  $x$  that satisfies the first-order necessary condition for optimality: for all feasible  $z \in \mathbf{R}^n$ ,  $(\nabla f(x), z - x) \geq 0$ . Equivalently,  $x$  is a Karush–Kuhn–Tucker point for problem (1). As in the case of unconstrained minimization, pattern search methods for bound constrained problems accomplish this without an explicit representation of the gradient or the directional derivative. In particular, we prove global convergence in the bound constrained case even though pattern search methods do not have explicit information concerning the gradient and its projection onto the feasible region and consequently do not explicitly enforce a notion of sufficient feasible decrease.

The general specification of pattern search methods for bound constrained minimization gives one broad latitude in designing such algorithms. Moreover, as we shall discuss, classical pattern search methods for unconstrained minimization—such as *coordinate search* with fixed step sizes and the original *pattern search* of Hooke and Jeeves [9]—can be generalized without modification to the bound constrained case. We also will show that not all pattern search methods for unconstrained minimization immediately generalize to bound constrained problems: in §5.2 we will present a counterexample that defeats G.E.P. Box’s method of *evolutionary operation* using two-level factorial designs [1], [3, 12] and show how the convergence theory guides us to a remedy that uses *composite designs* [2], instead of the simpler factorial or fractional factorial designs. The *multidirectional search algorithm* of Dennis and Torczon [7, 13] also requires us to augment the pattern used for the algorithm; again we find a straightforward extension, but one that reveals much about the interesting behavior of the simplices which characterize that method.

**Notation.** We denote by  $\mathbf{R}$ ,  $\mathbf{Q}$ ,  $\mathbf{Z}$ , and  $\mathbf{N}$  the sets of real, rational, integer, and natural numbers, respectively.

Unless otherwise noted, norms are assumed to be the Euclidean norm. The feasible region for problem (1) we denote by  $\Omega$ :

$$\Omega = \{ x \in \mathbf{R}^n \mid \ell \leq x \leq u \}.$$

The projection onto  $\Omega$  we denote by  $P$ . If for scalar  $t$  we define

$$p_j(t) = \begin{cases} \ell_j & \text{if } t < \ell_j \\ t & \text{if } \ell_j \leq t \leq u_j \\ u_j & \text{if } t > u_j, \end{cases}$$

then the projection of  $x = (x_1, \dots, x_n)^T$  is given by

$$P(x) = \sum_{j=1}^n p_j(x_j) e_j,$$

where  $\{e_j\}$ ,  $j = 1, \dots, n$  are the standard basis vectors. On those few occasions where we must denote components of subscripted vectors, we use the following notation:  $q_{k,j}$  denotes the  $j^{\text{th}}$  component of the vector  $q_k$ .

We will denote by  $g(x)$  the gradient  $\nabla f(x)$  of the objective. Finally, let

$$L_\Omega(y) = \{ x \in \Omega \mid f(x) \leq f(y) \}.$$

**2. Pattern Search Methods.** We begin by defining the general pattern search method for the bound constrained problem (1); it differs from that for unconstrained problems [14] in only a few particulars, which we summarize in §2.5.

**2.1. The Pattern.** As with pattern search methods for unconstrained problems, to define a pattern we need two components: a *basis matrix* and a *generating matrix*.

The basis matrix is a nonsingular matrix  $B \in \mathbf{R}^{n \times n}$ .

The generating matrix is a matrix  $C_k \in \mathbf{Z}^{n \times p}$ , where  $p > 2n$ . We partition the generating matrix into components

$$(2) \quad C_k = \begin{bmatrix} M_k & -M_k & L_k \end{bmatrix} = \begin{bmatrix} \Gamma_k & L_k \end{bmatrix}.$$

We require that  $M_k \in \mathbf{M} \subset \mathbf{Z}^{n \times n}$ , where  $\mathbf{M}$  is a finite set of nonsingular matrices, and that  $L_k \in \mathbf{Z}^{n \times (p-2n)}$  and that it contains at least one column, a column of zeroes.

A *pattern*  $P_k$  is then defined by the columns of the matrix  $P_k = BC_k$ . For convenience, we use the partition of the generating matrix  $C_k$  given in (2) to partition  $P_k$  as follows:

$$P_k = BC_k = \begin{bmatrix} BM_k & -BM_k & BL_k \end{bmatrix} = \begin{bmatrix} B\Gamma_k & BL_k \end{bmatrix}.$$

We also require the matrix  $BM_k$  to be diagonal:

$$(3) \quad BM_k = \text{diag}(d_k^i), \quad i = 1, \dots, n.$$

This condition, absent in the case of unconstrained minimization, is needed in order to ensure that we can find feasible points in the pattern that will also produce decrease in the objective. As we shall see, this condition is not especially restrictive and is satisfied by all of the commonly encountered pattern search algorithms or straightforward variants of them.

Given  $\Delta_k \in \mathbf{R}$ ,  $\Delta_k > 0$ , we define a *trial step*  $s_k^i$  to be any vector of the form  $s_k^i = \Delta_k B c_k^i$ , where  $c_k^i$  denotes a column of  $C_k = [c_k^1 \cdots c_k^p]$ . We call a trial step  $s_k^i$  *feasible* if  $(x_k + s_k^i) \in \Omega$ . At iteration  $k$ , a *trial point* is any point of the form  $x_k^i = x_k + s_k^i$ , where  $x_k$  is the current iterate.

**2.2. The Bound Constrained Exploratory Moves.** Pattern search methods proceed by conducting a series of *exploratory moves* about the current iterate  $x_k$  to choose a new iterate  $x_{k+1} = x_k + s_k$ , for some feasible step  $s_k$  determined during the course of the exploratory moves. The following Hypotheses on Bound Constrained Exploratory Moves allows a broad choice of exploratory moves while ensuring the properties required to prove convergence. By abuse of notation, if  $A$  is a matrix,  $y \in A$  means that the vector  $y$  is a column of  $A$ .

**Hypotheses on Bound Constrained Exploratory Moves.**

1.  $s_k \in \Delta_k P_k \equiv \Delta_k B C_k \equiv \Delta_k [B \Gamma_k \ B L_k]$ .
2.  $(x_k + s_k) \in \Omega$ .
3. If  $\min \{ f(x_k + y) \mid y \in \Delta_k B \Gamma_k, x_k + y \in \Omega \} < f(x_k)$ ,  
then  $f(x_k + s_k) < f(x_k)$ .

**2.3. The Generalized Pattern Search Method.** Algorithm 1 states the generalized pattern search method for minimization with bound constraints. To define a particular pattern search method, we must specify the basis matrix  $B$ , the generating matrix  $C_k$ , the bound constrained exploratory moves to be used to produce a feasible step  $s_k$ , and the algorithms for updating  $C_k$  and  $\Delta_k$ .

**Algorithm 1.** The Generalized Pattern Search Method for Bound Constrained Problems.

Let  $x_0 \in \Omega$  and  $\Delta_0 > 0$  be given.

For  $k = 0, 1, \dots$ ,

- a) Compute  $f(x_k)$ .
- b) Determine a step  $s_k$  using a bound constrained exploratory moves algorithm.
- c) Compute  $\rho_k = f(x_k) - f(x_k + s_k)$ .
- d) If  $\rho_k > 0$  then  $x_{k+1} = x_k + s_k$ . Otherwise  $x_{k+1} = x_k$ .
- e) Update  $C_k$  and  $\Delta_k$ .

**2.4. The Updates.** Algorithm 2 specifies the requirements for updating  $\Delta_k$ . The aim of the update of  $\Delta_k$  is to force  $\rho_k > 0$ . An iteration with  $\rho_k > 0$  is *successful*; otherwise, the iteration is *unsuccessful*. Note that to accept a step we only require *simple*, as opposed to *sufficient*, decrease.

**Algorithm 2.** Updating  $\Delta_k$ .

Let  $\tau \in \mathbf{Q}$ ,  $\tau > 1$ , and  $\{w_0, w_1, \dots, w_L\} \subset \mathbf{Z}$ ,  $w_0 < 0$ , and  $w_i \geq 0$ ,  $i = 1, \dots, L$ . Let  $\theta = \tau^{w_0}$ , and  $\lambda_k \in \Lambda = \{\tau^{w_1}, \dots, \tau^{w_L}\}$ .

- a) If  $\rho_k \leq 0$  then  $\Delta_{k+1} = \theta \Delta_k$ .
- b) If  $\rho_k > 0$  then  $\Delta_{k+1} = \lambda_k \Delta_k$ .

The conditions on  $\theta$  and  $\Lambda$  ensure that  $0 < \theta < 1$  and  $\lambda_i \geq 1$  for all  $\lambda_i \in \Lambda$ . Thus, if an iteration is successful it may be possible to increase the step length parameter  $\Delta_k$ , but  $\Delta_k$  is not allowed to decrease.

**2.5. Differences between Pattern Search Methods for Unconstrained and Bound Constrained Minimization.** There are only two additional restrictions required of pattern search methods to ensure convergence for the bound constrained case.

First, we note that pattern search methods are *feasible point* methods; the search begins with a point that satisfies the bounds and maintains feasibility throughout the search. This can be seen in Algorithm 1, where we require  $x_0 \in \Omega$ . This requirement also appears in the Hypotheses on Bound Constrained Exploratory Moves: if simple decrease on the function value at the current iterate can be found among any of the feasible trial steps contained in the columns of  $\Delta_k B \Gamma_k$ , then the exploratory moves must produce a feasible step  $s_k$  that also gives simple decrease on the function value at the current iterate.

The second, and more interesting, restriction is that the *core pattern*  $BM_k$  must be defined by a diagonal matrix. Because the columns of the pattern matrix determine the directions of the steps that may be considered, we need to ensure that if we are not at a constrained stationary point, we have at least one feasible direction of descent. Practically, we must ensure that we have directions that allow us to move parallel to the constraints. Requiring  $BM_k$  to be a diagonal matrix is sufficient. And as we will show in §5.2, this requirement is unavoidable.

We note an equivalence between pattern search methods for bound constrained problems and an exact penalization approach to problem (1). Applying a pattern search method for problem (1) produces exactly the same iterates as applying such an algorithm to the unconstrained problem

$$\text{minimize } F(x),$$

where

$$(4) \quad F(x) = \begin{cases} f(x) & \text{if } x \in \Omega \\ \infty & \text{otherwise.} \end{cases}$$

In fact, this is one classical approach used with direct search methods to ensure that the iterates produced remain feasible (see, for instance, [10, 11]). In the case of pattern search methods this formulation is not simply a conceptual approach; pattern search methods are directly applicable to this exact penalty function since they do not rely on derivatives. However, as we will demonstrate in §5.2, this exact penalization approach cannot be applied with an arbitrary pattern search method for unconstrained minimization; we must require that  $BM_k$  be diagonal.

**2.6. Results from the Unconstrained Theory.** We recall the following results from [14], to which we refer the reader for the proofs. The first result indicates one sense in which  $\Delta_k$  regulates step length.

LEMMA 2.1 (LEMMA 3.1 FROM [14]). *There exists a constant  $\zeta_* > 0$ , independent of  $k$ , such that for any trial step  $s_k^i \neq 0$  produced by a generalized pattern search method (Algorithm 1) we have  $\|s_k^i\| \geq \zeta_* \Delta_k$ .*

The next result is key to the convergence of pattern search methods. It states that the iterates produced by a pattern search method have a rigid algebraic structure.

**THEOREM 2.2 (THEOREM 3.2 FROM [14]).** Any iterate  $x_N$  produced by a generalized pattern search method (Algorithm 1) can be expressed in the following form:

$$(5) \quad x_N = x_0 + (\beta^{r_{LB}} \alpha^{-r_{UB}}) \Delta_0 B \sum_{k=0}^{N-1} z_k,$$

where

- $x_0$  is the initial guess,
- $\beta/\alpha \equiv \tau$ , with  $\alpha, \beta \in \mathbf{N}$  and relatively prime, and  $\tau$  is as defined in the algorithm for updating  $\Delta_k$  (Algorithm 2),
- $r_{LB}$  and  $r_{UB}$  depend on  $N$ ,
- $\Delta_0$  is the initial choice for the step length control parameter,
- $B$  is the basis matrix, and
- $z_k \in \mathbf{Z}^n$ ,  $k = 0, \dots, N-1$ .

The last result we recollect says, in conjunction with Lemma 2.1, that if we bound the size of the elements of the generating matrix (which is a reasonable thing to do), then  $\Delta_k$  completely regulates the size of the steps a pattern search method takes.

**LEMMA 2.3 (LEMMA 3.6 FROM [14]).** If there exists a constant  $C > 0$  such that for all  $k$ ,  $C > \|c_k^i\|$ , for all  $i = 1, \dots, p$ , then there exists a constant  $\psi_* > 0$ , independent of  $k$ , such that for any trial step  $s_k^i$  produced by a generalized pattern search method (Algorithm 1) we have  $\Delta_k \geq \psi_* \|s_k^i\|$ .

**3. Convergence Theory.** We now present the first-order constrained stationary point convergence theory for pattern search methods for bound constrained problems. We begin by defining, for feasible  $x$ , the quantity

$$q(x) = P(x - g(x)) - x.$$

In the bound constrained theory the quantity  $q(x)$  plays the role of  $g(x)$  in the unconstrained theory, giving us a continuous measure of how close we are to constrained stationarity, as in the theory for methods based explicitly on derivatives (e.g., [6], where we got the idea). The following proposition summarizes properties of  $q$  that we will shortly need, particularly the fact that  $x$  is a constrained stationary point for (1) if and only if  $q(x) = 0$ . While stated for the particular domain  $\Omega$ , the proposition holds for any closed convex domain.

**PROPOSITION 3.1.** Let  $x \in \Omega$ . Then

$$(6) \quad \|q(x)\| \leq \|g(x)\|,$$

$$(7) \quad \|q(x)\| \leq \|P(g(x))\|,$$

and

$$(8) \quad x \text{ is a stationary point for problem (1) if and only if } q(x) = 0.$$

*Proof.* We need the following properties of the projection:

- (a) If  $z \in \Omega$  then  $(P(x) - x, z - P(x)) \geq 0$  for all  $x \in \mathbf{R}^n$ .
- (b) For all  $x, y \in \mathbf{R}^n$ ,  $\|P(y) - P(x)\| \leq \|y - x\|$ .
- (c) If  $T(x)$  is the tangent cone of  $\Omega$  at  $x$ , then

$$\min \{ (g(x), v) \mid v \in T(x), \|v\| \leq 1 \} = -\|P(g(x))\|.$$

The proofs of (a) and (b) may be found in [8], and the proof of (c) may be found in [5] (there is a difference in sign between  $P(g(x))$  here and the quantity  $\nabla_{\Omega} f(x)$  as it is defined in the latter reference).

We first prove (6). If  $x \in \Omega$ , then  $P(x) = x$  and so from (b),

$$\|q(x)\| = \|P(x - g(x)) - P(x)\| \leq \|(x - g(x)) - x\| = \|g(x)\|,$$

which is (6).

Next we establish (7). Since  $P(x - g(x)) - x \in T(x)$ , we obtain from (c) the estimate

$$(g(x), P(x - g(x)) - x) \geq -\|P(g(x))\| \|P(x - g(x)) - x\|$$

or

$$(9) \quad (g(x), q(x)) \geq -\|P(g(x))\| \|q(x)\|.$$

Meanwhile, from (a) we obtain

$$(q(x) + g(x), -q(x)) = (P(x - g(x)) - (x - g(x)), x - P(x - g(x))) \geq 0,$$

whence

$$(10) \quad (g(x), q(x)) \leq -\|q(x)\|^2.$$

Combining (9) and (10) we arrive at

$$\|q(x)\| \leq \|P(g(x))\|,$$

which is (7).

Now consider (8). First suppose that  $x$  is a stationary point for problem (1). Then  $P(g(x)) = 0$ , so from (7) it immediately follows that  $q(x) = 0$ . Conversely, suppose that  $q(x) = 0$ . Then  $P(x - g(x)) = x$ , so from (a) we have, for all  $z \in \Omega$ ,

$$(P(x - g(x)) - (x - g(x)), z - P(x - g(x))) \geq 0,$$

or

$$(g(x), z - x) \geq 0.$$

But this is nothing other than the first-order necessary condition for optimality for problem (1) at  $x$ .  $\square$



We can now state the first convergence result for the general pattern search method for bound constrained minimization.

**THEOREM 3.2.** *Let  $L_\Omega(x_0)$  be compact and suppose  $f$  is continuously differentiable on  $L_\Omega(x_0)$ . Let  $\{x_k\}$  be the sequence of iterates produced by a generalized pattern search method for bound constrained minimization (Algorithm 1). Then*

$$\liminf_{k \rightarrow +\infty} \|q(x_k)\| = 0.$$

The proof of this theorem is given in §4.1, after we have established the necessary intermediate results.

We can strengthen the result given in Theorem 3.2 in the same way that we do in the unconstrained case [14]. First, we require the columns of the generating matrix  $C_k$  to remain bounded in norm, i.e., that there exists a constant  $C > 0$  such that for all  $k$ ,  $C > \|c_k^i\|$ , for all  $i = 1, \dots, p$ . Second, we replace the original Hypotheses on Bound Constrained Exploratory Moves with a stronger version, given below. Third, we require that  $\lim_{k \rightarrow +\infty} \Delta_k = 0$ . All the algorithms described in §5, except multidirectional search, satisfy this third condition because of the customary choice of  $\Lambda = \{1\} \equiv \{\tau^0\}$ . However, it is not necessary to force the steps to be non-increasing.

**Strong Hypotheses on Bound Constrained Exploratory Moves.**

1.  $s_k \in \Delta_k P_k \equiv \Delta_k B C_k \equiv \Delta_k [B \Gamma_k B L_k]$ .
2.  $(x_k + s_k) \in \Omega$ .
3. If  $\min \{ f(x_k + y) \mid y \in \Delta_k B \Gamma_k, x_k + y \in \Omega \} < f(x_k)$ ,  
then  $f(x_k + s_k) < \min \{ f(x_k + y) \mid y \in \Delta_k B \Gamma_k, x_k + y \in \Omega \}$ .

**THEOREM 3.3.** *Let  $L_\Omega(x_0)$  be compact and suppose  $f$  is continuously differentiable on  $L_\Omega(x_0)$ . In addition, assume that the columns of the generating matrices are uniformly bounded in norm, that  $\lim_{k \rightarrow +\infty} \Delta_k = 0$ , and that the generalized pattern search method for bound constrained minimization (Algorithm 1) enforces the Strong Hypotheses on Bound Constrained Exploratory Moves. Then for the sequence of iterates  $\{x_k\}$  produced by the generalized pattern search method for bound constrained minimization,*

$$\lim_{k \rightarrow +\infty} \|q(x_k)\| = 0.$$

The proof will be found in §4.2.

**4. Proof of Theorems 3.2 and 3.3.** Given an iterate  $x_k$ , let  $g_k = g(x_k)$  and  $q_k = q(x_k)$ . Let  $\omega$  denote the following modulus of continuity of  $g$ : given  $x \in \mathbf{R}^n$  and  $\varepsilon > 0$ ,

$$\omega(x, \varepsilon) = \sup \{ \delta > 0 \mid \|g(y) - g(x)\| < \varepsilon \text{ for all } y \text{ such that } \|y - x\| < \delta \}.$$

We begin with an elementary proposition concerning descent directions.

PROPOSITION 4.1. Let  $d \in \mathbf{R}^n$  and suppose that  $f$  restricted to the closed line segment connecting  $x$  and  $x+d$  is continuously differentiable. Assume, too, that  $g(x) \neq 0$  and  $g(x)^T d \leq -\varepsilon \|d\|$ . Then, if  $\|d\| < \omega(x, \frac{\varepsilon}{2})$ ,

$$f(x+d) - f(x) \leq -\frac{\varepsilon}{2} \|d\|.$$

*Proof.* From the mean-value theorem, we have, for some  $y$  on the line segment between  $x$  and  $x+d$ ,

$$\begin{aligned} f(x+d) - f(x) &= g(x)^T d + (g(y) - g(x))^T d \\ &\leq -\varepsilon \|d\| + \|g(y) - g(x)\| \|d\|. \end{aligned}$$

If  $\|d\| < \omega(x, \frac{\varepsilon}{2})$ , then  $\|g(y) - g(x)\| \leq \frac{\varepsilon}{2}$  and the result follows.  $\square$

In the proof of the next result the bound constrained and the unconstrained cases most differ. The proof of Proposition 4.2 implicitly relies on the fact that in the bound constrained case, the directions in the pattern defined by the columns of  $BM_k$  are coordinate directions and thus are oriented normal and tangent to the faces of the feasible region. That this is not merely convenient is made clear by the counterexample in §5.2.

PROPOSITION 4.2. Suppose that  $f$  is continuously differentiable on  $L_\Omega(x_0)$  and  $q(x_k) \neq 0$ . Then there exists a  $\nu_k > 0$  such that if  $\Delta_k < \nu_k$ , then there is a trial step  $s_k^i$  defined by a column of  $\Delta_k B \Gamma_k$  for which  $(x_k + s_k^i) \in \Omega$  and

$$g_k^T s_k^i \leq -c_n \|q_k\| \|s_k^i\|,$$

where  $c_n = n^{-\frac{1}{2}}$ .

*Proof.* We restrict our attention to the steps defined by the columns of  $\Delta_k B \Gamma_k$ ; by hypothesis,  $\Delta_k B \Gamma_k \equiv \Delta_k B [M_k - M_k] = \Delta_k [\text{diag}(d_k^i) - \text{diag}(d_k^i)]$  (see (3)). Choose an index  $m$  for which

$$(11) \quad |q_{k,m}| = \|q_k\|_\infty \geq \frac{1}{\sqrt{n}} \|q_k\|,$$

where  $q_{k,m}$  is the  $m^{\text{th}}$  component of  $q_k$ . Note that it is also the case that

$$(12) \quad |g_{k,m}| \geq |q_{k,m}|$$

and  $\text{sign}(g_{k,m}) = \text{sign}(q_{k,m})$ .

Let  $s_k^i = -\text{sign}(g_{k,m}) \Delta_k |d_k^m| e_m$ ; this vector will be among the columns of  $\Delta_k B \Gamma_k$ . Since  $x_k + q_k = P(x_k - g_k)$  is feasible, we have  $\ell \leq x_k + q_k \leq u$  and thus

$$\ell_m \leq x_{k,m} + q_{k,m} \leq u_m.$$

It follows that if  $\Delta_k |d_k^m| \leq |q_{k,m}|$ , then the trial point  $x_k^i = x_k + s_k^i$  will be feasible. Moreover, from (11) and (12),

$$g_k^T s_k^i = -\text{sign}(g_{k,m}) \Delta_k |d_k^m| |g_{k,m}| = -\frac{1}{\sqrt{n}} \|s_k^i\| \|g_{k,m}\| \leq -\frac{1}{\sqrt{n}} \|s_k^i\| \|q_k\|.$$

Defining  $\nu_k = \|q_k\|_\infty / |d_k^m|$  then does the trick.  $\square$

PROPOSITION 4.3. Suppose that  $L_\Omega(x_0)$  is compact and that  $f$  is continuously differentiable on  $L_\Omega(x_0)$ . Then given any  $\eta > 0$ , there exists  $\delta > 0$ , independent of  $k$ , such that if  $\Delta_k < \delta$  and  $\|q(x_k)\| > \eta$ , the pattern search method for bound constrained minimization will find an acceptable step  $s_k$ ; i.e.,  $f(x_k + s_k) < f(x_k)$  and  $(x_k + s_k) \in \Omega$ .

If, in addition, the columns of the generating matrix remain bounded in norm and we enforce the Strong Hypotheses on Bound Constrained Exploratory Moves, then, given any  $\eta > 0$ , there exist  $\delta > 0$  and  $\sigma > 0$ , independent of  $k$ , such that if  $\Delta_k < \delta$  and  $\|q(x_k)\| > \eta$ , then

$$f(x_{k+1}) \leq f(x_k) - \sigma \|q(x_k)\| \|s_k\|.$$

Proof. Since  $g(x)$  is uniformly continuous on  $L_\Omega(x_0)$ , there exists  $\omega_* > 0$  such that

$$\omega(x_k, n^{-\frac{1}{2}}\eta) \geq \omega_*$$

for all  $k$  for which  $\|q_k\| > \eta$ .

Next, choose  $d^* > 0$  such that  $d_k^i \leq d^*$  for all  $i$  and  $k$ . This we can do because the set  $\{d_k^i\}$  is finite (see (3) and the conditions on  $M_k$  given in §2.1). Let

$$\nu_* = \frac{\eta}{n^{\frac{1}{2}}d^*};$$

then

$$\nu_* = \frac{\eta}{n^{\frac{1}{2}}d^*} \leq \frac{\|q_k\|}{n^{\frac{1}{2}}d^*} \leq \frac{\|q_k\|_\infty}{d^*} \leq \nu_k$$

for all  $k$  for which  $\|q_k\| > \eta$ , where  $\nu_k$  is as in Proposition 4.2.

Finally, let

$$\delta = \min(\nu_*, \omega_*/d^*).$$

Now suppose  $\|q(x_k)\| > \eta$  and  $\Delta_k < \delta$ . Since  $\Delta_k < \nu_k$ , Proposition 4.2 assures us of the existence of a step  $s_k^i$  defined by a column of  $\Delta_k B \Gamma_k$  such that  $(x_k + s_k^i) \in \Omega$  and

$$g_k^T s_k^i \leq -c_n \|q_k\| \|s_k^i\|.$$

At the same time, we also have

$$\|s_k^i\| \leq \Delta_k d^* \leq \omega_* \leq \omega(x_k, c_n \|q_k\|).$$

So by Proposition 4.1,

$$f(x_k + s_k^i) - f(x_k) \leq -\frac{c_n}{2} \|q_k\| \|s_k^i\|.$$

Thus, when  $\Delta_k < \delta$ ,  $f(x_k^i) \equiv f(x_k + s_k^i) < f(x_k)$  for at least one feasible  $s_k^i \in \Delta_k B \Gamma_k$ . The Hypotheses on Bound Constrained Exploratory Moves guarantee that if

$$\min \{ f(x_k + y) \mid y \in \Delta_k B \Gamma_k, x_k + y \in \Omega \} < f(x_k),$$

then  $f(x_k + s_k) < f(x_k)$  and  $(x_k + s_k) \in \Omega$ . This proves the first part of the Proposition.

If, in addition, we enforce the Strong Hypotheses on Bound Constrained Exploratory Moves, then we actually have

$$f(x_{k+1}) - f(x_k) \leq -\frac{c_n}{2} \|q_k\| \|s_k^i\|.$$

Lemma 2.1 then ensures that

$$f(x_{k+1}) \leq f(x_k) - \frac{c_n}{2} \zeta_* \Delta_k \|q(x_k)\|.$$

Applying Lemma 2.3, we arrive at

$$f(x_{k+1}) \leq f(x_k) - \sigma \|q(x_k)\| \|s_k\|,$$

where  $\sigma = \frac{c_n}{2} \zeta_* \psi_*$ .  $\square$

**COROLLARY 4.4.** *Suppose that  $L_\Omega(x_0)$  is compact and that  $f$  is continuously differentiable on  $L_\Omega(x_0)$ . Suppose, too, that  $\liminf_{k \rightarrow +\infty} \|q(x_k)\| \neq 0$ . Then there exists a constant  $\Delta_* > 0$  such that for all  $k$ ,  $\Delta_k > \Delta_*$ .*

*Proof.* By hypothesis, there exists  $N$  and  $\eta > 0$  such that for all  $k > N$ ,  $\|q(x_k)\| > \eta$ . By Proposition 4.3, we can find  $\delta$  such that if  $k > N$  and  $\Delta_k < \delta$ , then we will find an acceptable step. In view of the algorithm for updating  $\Delta_k$  in §2.4, we are assured that for all  $k > N$ ,  $\Delta_k > \theta\delta$ . We may then take  $\Delta_* = \min\{\Delta_0, \dots, \Delta_N, \theta\delta\}$ .  $\square$

The next theorem combines the strict algebraic structure of the iterates with the simple decrease condition of the generalized pattern search algorithm for bound constrained problems, along with the algorithm for updating  $\Delta_k$ , to give us a useful fact about the limiting behavior of  $\Delta_k$ .

**THEOREM 4.5.** *Assume that  $L_\Omega(x_0)$  is compact. Then  $\liminf_{k \rightarrow +\infty} \Delta_k = 0$ .*

*Proof.* The proof is like that of Theorem 3.3 in [14]. Suppose  $0 < \Delta_{LB} \leq \Delta_k$  for all  $k$ . Using the algorithm for updating  $\Delta_k$  found in §2.4, it is possible to write  $\Delta_k$  as  $\Delta_k = \tau^{r_k} \Delta_0$ , where  $r_k \in \mathbf{Z}$ .

The hypothesis that  $\Delta_{LB} \leq \Delta_k$  for all  $k$  means that the sequence  $\{\tau^{r_k}\}$  is bounded away from zero. Meanwhile, we also know that the sequence  $\{\Delta_k\}$  is bounded above because all the iterates  $x_k$  must lie inside the set  $L_\Omega(x_0) = \{x \in \Omega : f(x) \leq f(x_0)\}$  and the latter set is compact; Lemma 2.1 then guarantees an upper bound  $\Delta_{UB}$  for  $\{\Delta_k\}$ . This, in turn, means that the sequence  $\{\tau^{r_k}\}$  is bounded above. Consequently, the sequence  $\{\tau^{r_k}\}$  is a finite set. Equivalently, the sequence  $\{r_k\}$  is bounded above and below.

Next we recall the exact identity of the quantities  $r_{LB}$  and  $r_{UB}$  in Theorem 2.2; the details are found in the proof of Theorem 3.3 in [14]. In the context of Theorem 2.2,

$$r_{LB} = \min_{0 \leq k < N} \{r_k\} \quad r_{UB} = \max_{0 \leq k < N} \{r_k\}.$$

If, in the matter at hand, we let

$$(13) \quad r_{LB} = \min_{0 \leq k < +\infty} \{r_k\} \quad r_{UB} = \max_{0 \leq k < +\infty} \{r_k\},$$

then (5) holds for the bounds given in (13), and we see that for all  $k$ ,  $x_k$  lies in the translated integer lattice  $G$  generated by  $x_0$  and the columns of  $\beta^{r_{LB}} \alpha^{-r_{UB}} \Delta_0 B$ .

The intersection of the compact set  $L_\Omega(x_0)$  with the lattice  $G$  is finite. Thus, there must exist at least one point  $x_*$  in the lattice for which  $x_k = x_*$  for infinitely many  $k$ .

We now appeal to the simple decrease condition in Algorithm 1 (d), which guarantees that an iterate cannot be revisited infinitely many times since we accept a new step  $s_k$  if and only if  $f(x_k) > f(x_k + s_k)$  and  $(x_k + s_k) \in \Omega$ . Thus there exists an  $N$  such that for all  $k \geq N$ ,  $x_k = x_*$ , which implies that  $\rho_k = 0$ .

We now appeal to the algorithm for updating  $\Delta_k$  (Algorithm 2 (a)) to see that  $\Delta_k \rightarrow 0$ , thus leading to a contradiction.  $\square$

**4.1. The Proof of Theorem 3.2.** The proof is like that of Theorem 3.5 in [14]. Suppose that  $\liminf_{k \rightarrow +\infty} \|q(x_k)\| \neq 0$ . Then Corollary 4.4 tells us that there exists  $\Delta_* > 0$  such that for all  $k$ ,  $\Delta_k \geq \Delta_*$ . But this contradicts Theorem 4.5.

**4.2. The Proof of Theorem 3.3.** The proof, also by contradiction, follows that of Theorem 3.7 in [14]. Suppose  $\limsup_{k \rightarrow +\infty} \|q(x_k)\| \neq 0$ . Let  $\varepsilon > 0$  be such that there exists a subsequence  $\|q(x_{m_i})\| \geq \varepsilon$ . Since

$$\liminf_{k \rightarrow +\infty} \|q(x_k)\| = 0,$$

given any  $0 < \eta < \varepsilon$ , there exists an associated subsequence  $l_i$  such that

$$\|q(x_k)\| > \eta \quad \text{for} \quad m_i \leq k < l_i, \quad \|q(x_{l_i})\| < \eta.$$

Since  $\Delta_k \rightarrow 0$ , we can appeal to Proposition 4.3 to obtain for  $m_i \leq k < l_i$ ,  $i$  sufficiently large,

$$f(x_k) - f(x_{k+1}) \geq \sigma \|q(x_k)\| \|s_k\| \geq \sigma \eta \|s_k\|,$$

where  $\sigma > 0$ . Then the telescoping sum:

$$(f(x_{m_i}) - f(x_{m_i+1})) + (f(x_{m_i+1}) - f(x_{m_i+2})) + \cdots + (f(x_{l_i-1}) - f(x_{l_i})) \geq \sum_{k=m_i}^{l_i} \sigma \eta \|s_k\|$$

gives us

$$f(x_{m_i}) - f(x_{l_i}) \geq \sum_{k=m_i}^{l_i} \sigma \eta \|s_k\| \geq c' \|x_{m_i} - x_{l_i}\|.$$

Since  $f$  is bounded below,  $f(x_{m_i}) - f(x_{l_i}) \rightarrow 0$  as  $i \rightarrow +\infty$ , so  $\|x_{m_i} - x_{l_i}\| \rightarrow 0$  as  $i \rightarrow +\infty$ . Then, because  $q$  is uniformly continuous,  $\|q(x_{m_i}) - q(x_{l_i})\| < \eta$ , for  $i$  sufficiently large. However,

$$(14) \quad \|q(x_{m_i})\| \leq \|q(x_{m_i}) - q(x_{l_i})\| + \|q(x_{l_i})\| \leq 2\eta.$$

Since (14) must hold for any  $\eta$ ,  $0 < \eta < \varepsilon$ , we have a contradiction (e.g., try  $\eta = \frac{\varepsilon}{4}$ ).

**5. Examples of Pattern Search Methods for Bound Constrained Minimization.** A section of [14] is devoted to showing that each of the following four algorithms:

- coordinate search with fixed step lengths,
- evolutionary operation using two-level factorial designs ([1] and [3, 12]),
- the original pattern search method of Hooke and Jeeves [9], and
- the multidirectional search algorithm of Dennis and Torczon ([7] and [13])

are pattern search methods for unconstrained minimization. In this section we will discuss how these algorithms may be extended to bound constrained problems. We shall see that coordinate search and the pattern search method of Hooke and Jeeves extend without modification to the bound constrained case. On the other hand, in the case of multidirectional search, we must require the initial basis matrix to be a diagonal matrix (in the unconstrained case, we can allow any nonsingular basis matrix); in addition, we must augment the columns of the generating matrix to ensure a sufficient set of search directions. In the case of evolutionary operation, we also must augment the columns of the generating matrix, which we do using a classical variant of factorial designs [2].

The difference between pattern search methods for unconstrained problems and bound constrained problems lies in the two additional conditions discussed in §2.5. First, pattern search methods for bound constrained problems must start with a feasible iterate and choose feasible trial steps. Second, the core pattern  $BM_k$  must be defined by a diagonal matrix.

We assume that we begin with a feasible iterate; by design pattern search methods for bound constrained problems thereafter accept only feasible iterates. Thus, the only thing we will really need to check is that the core pattern  $BM_k$  is defined by a diagonal matrix.

It is this latter condition that causes us to restrict the admissible choice of the basis matrix in multidirectional search and then augment the columns of the generating matrix. Moreover, G.E.P. Box's method of evolutionary operation using two-level factorial designs does not satisfy this diagonality condition; we will present a simple counterexample that show how evolutionary operation can fail as a consequence in the bound constrained case and propose a straightforward remedy for the problem.

**5.1. Coordinate Search and the Pattern Search Method of Hooke and Jeeves.** Coordinate search and the pattern search method of Hooke and Jeeves extend to bound constrained problems without change. In both cases the basis matrix  $B$  is typically chosen to be a diagonal matrix: either the identity or a matrix whose entries reflect the relative scaling of the variables. Furthermore, the first  $3^n$  columns of  $C_k$ , which are fixed for all iterations  $k$  of both algorithms, are composed of all possible combinations of  $\{-1, 0, 1\}$ . In [14] these columns are organized so that the first  $2n$  consist of the identity matrix  $I$  and its negative  $-I$ . In terms of our formalism, then,  $M_k = I$  for all iterations  $k$ . It follows that  $BM_k$  is a diagonal matrix, as required.

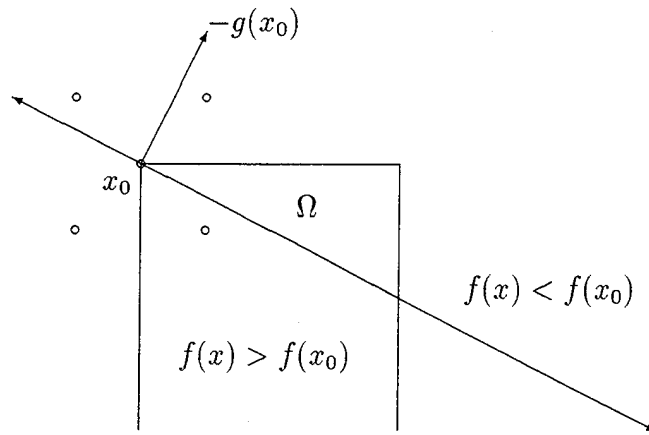


FIG. 1. An illustration of what can go wrong with factorial design in the bound constrained case.

**5.2. Evolutionary Operation Using Factorial Design.** For the evolutionary operation algorithm using factorial designs, the basis matrix  $B$  is usually selected to be the identity or a diagonal matrix chosen so that the entries along the diagonal represent the relative scaling among the variables. However, this convention is not sufficient to ensure that  $BM_k$  is a diagonal matrix.

The problem lies with the generating matrix  $C = [M \ -M \ L]$ . (The generating matrix  $C$  is fixed across all iterations of evolutionary operation.) The generating matrix contains in its columns all possible combinations of  $\{-1, 1\}$  to which is appended a column of zeroes. Clearly, no subset of  $n$  columns of  $C$  can be chosen to form a diagonal matrix  $M$ .

The need for such a requirement in the bound constrained case can be seen from the following example. Consider the pattern search method of evolutionary operation using two-level factorial designs applied to the problem

$$\begin{aligned} &\text{minimize} && f(x) = -(x_1 + 2x_2) \\ &\text{subject to} && 0 \leq x_1 \leq 1 \\ &&& x_2 \leq 0. \end{aligned}$$

Suppose that we take as our initial iterate the point  $(0, 0)$ ; the pattern for factorial design would consist of the points indicated by the open circles in Figure 1 (we show a pattern with  $\Delta_k = 1/4$ ). We see that for any choice of  $\Delta_k \leq 1$  there only ever will be one feasible step produced by the pattern: the step given by the point in the lower right-hand corner of the pattern—in the figure, the point  $(1/4, -1/4)$ —and that this step will only produce increase in  $f$ . So evolutionary operation will never move from its starting point.

One remedy can be found in related work by G.E.P. Box and K.B. Wilson [2]. This would be to use a composite design instead of the two-level factorial design usually proposed. An example of such a design—one that satisfies the requirements of the

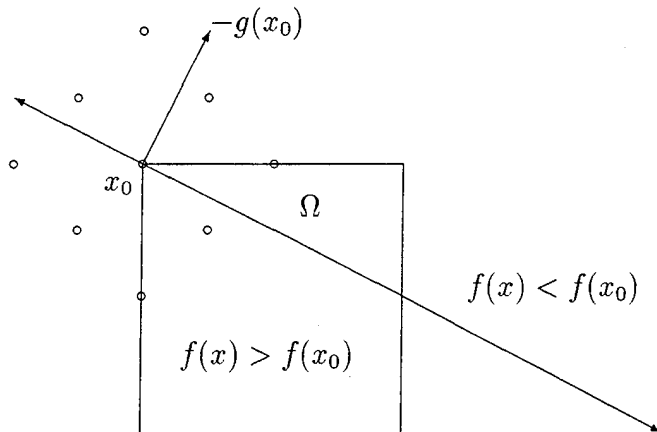


FIG. 2. An illustration of how the problem can be circumvented using a composite design.

bound constrained global convergence theory—chooses  $M$  to be the diagonal matrix with entries of 2 along the diagonal. With the additional  $2n$  columns that would then augment the original pattern, the algorithm would not fail for the example given above, as shown in Fig. 2. We now have a feasible step along the active constraint  $x_2 \leq 0$  that will produce descent for any choice of  $\Delta_k \leq 1/2$ .

As noted in our discussion of Proposition 4.2, by requiring  $BM_k$  to be a diagonal matrix, we are guaranteed that the directions in the core pattern are oriented normal and tangent to the faces of the feasible region. As our example illustrates, this requirement is essential.

**5.3. Multidirectional Search.** The reader should be forewarned that our description and discussion of multidirectional search takes a point of view that is ostensibly at odds with the formalism of §2.1. The generating matrix  $\Gamma$  is viewed as fixed; typically  $\Gamma = [M \ -M] \equiv [I \ -I]$ . The basis matrix, on the other hand, is viewed as varying from iteration to iteration so that  $B_k$  corresponds to the edges in the current simplex that are adjacent to the current iterate  $x_k$ . This is the reverse of the discussion in §2.1, where  $B$  is fixed and  $\Gamma_k$  varies. However, the former view of multidirectional search is not incompatible with the formalism of pattern search methods, as noted in [14], and as we shall have reason to discuss here.

The extension of multidirectional search to problems with bound constraints requires us to restrict the choice of a starting simplex and to augment the columns of the generating matrix.

The first restriction is minor and is usually satisfied by the customary choices made in practice. In multidirectional search, the columns of  $B_0$  are formed from the edges of an initial simplex adjacent to the initial iterate  $x_0$ . In the case of bound constraints, we restrict the starting simplex to be a right-angled simplex, i.e., the  $n + 1$  vertices of the simplex are  $x_0$  and the points  $x_0 + \alpha_i e_i$ , where  $\alpha_i \in \mathbf{R}$  and  $i = 1, \dots, n$ . Because of



this choice,  $B_0 = \text{diag}(\alpha_i)$ . Since  $M \equiv I$ , the product  $B_0M$  is a diagonal matrix.

However, even if the initial simplex is restricted to be a right-angled simplex so that  $B_0M$  is diagonal, there is no guarantee that in subsequent iterations  $B_kM$  will be diagonal. To understand why this is so, and how this may be corrected by augmenting the columns of the generating matrix, we need to discuss how multidirectional search fits within the formalism of pattern search methods. These details are absent from [14], so we present them here.

At iteration  $k$ , the basis matrix is

$$B_k = [b_k^1 \cdots b_k^n] = [(v_k^1 - v_k^0) \cdots (v_k^n - v_k^0)],$$

where  $v_k^i$ ,  $i = 0, \dots, n$  are the vertices of the simplex associated with multidirectional search at this iteration. Define

$$T_i = \begin{cases} I & i = 0 \\ -\left(I - e_i e_i^T - \sum_{m=1}^n e_i e_m^T\right) & i = 1, \dots, n. \end{cases}$$

Now consider what happens in the next iteration. If the iteration is unsuccessful, then  $v_{k+1}^0 = v_k^0$  and the new basis for the pattern, which is determined by the edges of the simplex emanating from  $v_{k+1}^0$ , is

$$B_{k+1} = B_k = B_k T_0.$$

If, on the other hand, the iteration is successful, then  $v_{k+1}^0 = v_k^0 - (v_k^j - v_k^0)$  for some  $j = 1, \dots, n$ , and the new basis will be the set of vectors

$$b_{k+1}^i = \begin{cases} b_k^j & \text{if } i = j \\ -b_k^i + b_k^j & \text{otherwise.} \end{cases}$$

In this case,

$$B_{k+1} = B_k T_j.$$

Thus, in general,

$$(15) \quad B_{k+1} = B_k T_{j_{k+1}},$$

and so

$$(16) \quad B_k = B_{k-1} T_{j_k} = B_{k-2} T_{j_{k-1}} T_{j_k} = \cdots = B_0 \prod_{i=1}^k T_{j_i}.$$

Our next goal is to simplify this relation further.

A short calculation shows that for  $i, \ell = 1, \dots, n$ ,

$$T_i T_\ell = I - e_\ell e_\ell^T - \sum_{m=1}^n e_\ell e_m^T - e_i e_i^T + \delta_\ell^i e_i e_\ell^T + \delta_\ell^i \sum_{m=1}^n e_i e_m^T + e_i e_\ell^T,$$

where  $\delta_\ell^i$  is the Kronecker delta. If  $i = \ell$ , this reduces to

$$(17) \quad T_i T_i = I$$

and if  $i \neq \ell$ ,

$$(18) \quad T_i T_\ell = I - e_\ell e_\ell^T - \sum_{m=1}^n e_\ell e_m^T - e_i e_i^T + e_i e_\ell^T = -T_\ell - e_i e_i^T + e_i e_\ell^T.$$

If  $i \neq \ell$ ,

$$(19) \quad T_\ell e_i = - \left( I - e_\ell e_\ell^T - \sum_{m=1}^n e_\ell e_m^T \right) e_i = e_\ell - e_i$$

and

$$(20) \quad (T_i (-T_\ell)) e_i = \left( -I + e_\ell e_\ell^T + \sum_{m=1}^n e_\ell e_m^T + e_i e_i^T - e_i e_\ell^T \right) e_i = e_\ell.$$

From (17), (18), and (19) we obtain

$$(21) \quad \begin{aligned} T_\ell T_i T_\ell &= T_\ell (-T_\ell - e_i e_i^T + e_i e_\ell^T) \\ &= -I - T_\ell e_i e_i^T + T_\ell e_i e_\ell^T \\ &= -I - (e_\ell - e_i) e_i^T + (e_\ell - e_i) e_\ell^T \\ &= - \left( I - e_i e_i^T - e_\ell e_\ell^T + e_\ell e_i^T + e_i e_\ell^T \right). \end{aligned}$$

This latter matrix we recognize as the negative of the elementary permutation matrix  $E(i, \ell)$  that swaps the  $i^{\text{th}}$  and  $\ell^{\text{th}}$  columns when acting on matrices from the right. Using (17) and (21), we obtain the rule

$$T_i T_\ell = \begin{cases} I & \text{if } i = \ell \\ -T_\ell E(i, \ell) & \text{otherwise.} \end{cases}$$

We can then use this formula to reduce (16) to

$$B_k = \pm B_0 T_{\ell_k} \Pi_k,$$

for some  $T_{\ell_k}$  and permutation matrix  $\Pi_k$ .

This relationship reveals several things. The first is that it reconciles the usual description of multidirectional search with the formal abstract definition of a pattern search method; the pattern matrix is given by

$$(22) \quad B_k C = \pm B_0 T_{j_k} \Pi_k [I - I \ 0] = B_0 [T_{j_k} \ -T_{j_k} \ 0] \Pi_k \equiv B C_k.$$

That is, we may interpret multidirectional search in terms of a fixed basis  $B$  and a changing generating matrix  $C_k$ .

We can also see that while  $B\Gamma_0$  will be diagonal, this diagonality may be lost in subsequent iterations. However, the form of the generic pattern from the unconstrained algorithm suggests one way to circumvent this problem in the bound constrained case. This remedy will, moreover, preserve the geometric interpretation of the pattern in multidirectional search in terms of a simplex.

First, if we ignore the permutation in (22), which only affects column ordering, the pattern at iteration  $k$  in the unconstrained case is given by

$$B_k C \equiv B C_k = B_0 [T_{j_k} \quad -T_{j_k} \quad 0].$$

Suppose we augment the columns of  $C$  to include all the  $T_i$ :

$$C = [-T_0 \quad -T_1 \quad \cdots \quad -T_n \quad 0].$$

At any iteration  $k$ , up to a column permutation, the basis matrix is the matrix  $B_k = \pm B T_{j_k}$ ,  $j_k \in \{0, \dots, n\}$ . When we then form the pattern  $P_k = \Delta_k B_k C$ , we have

$$P_k = \Delta_k B_k C = \Delta_k B [\pm T_{j_k} T_0 \quad \pm T_{j_k} T_1 \quad \cdots \quad \pm T_{j_k} T_n] \equiv \Delta_k B C_k.$$

Now note that (20) means that for  $j_k \neq l$  the  $j_k^{\text{th}}$  column of  $-T_{j_k} T_l$  is the  $l^{\text{th}}$  basis vector. Consequently, we are guaranteed that by a permutation of the columns of  $C_k$ ,

$$C_k = [I \quad -I \quad L_k] \equiv [\Gamma \quad L_k],$$

where  $L_k$  changes at each iteration, but  $\Gamma$  does not. Since we require the initial simplex to be a right-angled simplex, we may then be assured that  $B\Gamma = [\text{diag}(\alpha_i) \quad -\text{diag}(\alpha_i)]$ , as required.

Moreover, this augmentation of  $C$  and the search through its columns can be implemented in a way that preserves the relationship of the pattern to the moving simplex that characterizes multidirectional search. This is possible because the matrices  $T_i$ ,  $i = 0, \dots, n$  capture how the basis changes in association with a change of simplex. This is the gist of (15). The implications for any implementation of this modification to multidirectional search to handle bound constraints will appear elsewhere.

**6. Conclusion.** We have presented a reasonable extension of pattern search methods for unconstrained minimization to bound constrained problems. The extension is supported by a global convergence theory as strong as that for the unconstrained case. The generalization makes few additional requirements and as we have seen in §5, the classical pattern search methods for unconstrained minimization or straightforward variants thereof carry over to the bound constrained case.

The extension to bound constrained problems also points the way to handling general linear inequalities. General linear inequalities will require a sufficiently rich set of directions in the pattern so that at any face of the feasible region, one will have directions both normal and tangent to the constraints. This we will pursue elsewhere.

One issue we have not discussed is that of identifying active constraints, as in [4, 5]. One would wish to show that if the sequence  $\{x_k\}$  converges to a nondegenerate

stationary point  $x_*$ , then in a finite number of iterations the iterates  $x_k$  land on the constraints active at  $x_*$  and remain thereafter on those constraints.

There are three difficulties in proving such a result for pattern search methods for bound constrained minimization. The first is relatively minor. If the iterates  $x_k$  are to identify the active constraints for a stationary point on the boundary of the feasible region, we must ensure that the lattice manifest in Theorem 2.2 actually allows iterates to land on the boundary. This requires additional but straightforward conditions on such quantities as  $x_0, \tau, \Delta_0$ , and the pattern matrices  $P_k$ . A related but more subtle difficulty is that the relative sizes of the steps in the core pattern and the remaining points in the pattern must obey certain relations in order to ensure that the algorithm does not take a purely interior approach to a point on the boundary. This rules out, for instance, certain of the composite designs suggested by G.E.P. Box and K.B. Wilson [2].

The most serious obstacle, which remains to be overcome, is showing that ultimately the iterates will land on the active constraints and remain there. For algorithms such as those considered in [4, 5], this is not a problem because the explicit use of the gradient impels the iterates to do this in the neighborhood of a nondegenerate stationary point. However, pattern search methods do not have this information. On the other hand, the kinship of pattern search methods and gradient projection methods makes us hopeful that ultimately we will be able to prove that pattern search methods also identify the active constraints in a finite number of iterations.

## REFERENCES

- [1] G. E. P. BOX, *Evolutionary operation: A method for increasing industrial productivity*, Applied Statistics, 6 (1957), pp. 81–101.
- [2] G. E. P. BOX AND K. B. WILSON, *On the experimental attainment of optimum conditions*, Journal of the Royal Statistical Society, Series B, XIII (1951), pp. 1–45.
- [3] M. J. BOX, D. DAVIES, AND W. H. SWANN, *Non-Linear Optimization Techniques*, ICI Monograph No. 5, Oliver & Boyd, Edinburgh, 1969.
- [4] J. V. BURKE AND J. J. MORÉ, *On the identification of active constraints*, SIAM Journal on Numerical Analysis, 25 (1988), pp. 1197–1211.
- [5] P. H. CALAMAI AND J. J. MORÉ, *Projected gradient methods for linearly constrained problems*, Mathematical Programming, 39 (1987), pp. 93–116.
- [6] A. R. CONN, N. I. M. GOULD, AND P. L. TOINT, *Global convergence of a class of trust region algorithms with simple bounds*, SIAM Journal on Numerical Analysis, 25 (1988), pp. 433–460.
- [7] J. E. DENNIS, JR. AND V. TORCZON, *Direct search methods on parallel machines*, SIAM J. on Optimization, 1 (1991), pp. 448–474.
- [8] J. C. DUNN, *Global and asymptotic convergence rate estimates for a class of projected gradient processes*, SIAM Journal on Control and Optimization, 19 (1981), pp. 368–400.
- [9] R. HOOKE AND T. A. JEEVES, *“Direct search” solution of numerical and statistical problems*, J. Assoc. Comput. Mach., 8 (1961), pp. 212–229.
- [10] J. A. NELDER AND R. MEAD, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.
- [11] W. SPENDLEY, G. R. HEXT, AND F. R. HIMSWORTH, *Sequential application of simplex designs in optimisation and evolutionary operation*, Technometrics, 4 (1962), pp. 441–461.
- [12] W. H. SWANN, *Direct search methods*, in Numerical Methods for Unconstrained Optimization, W. Murray, ed., Academic Press, 1972, pp. 13–28.

- [13] V. TORCZON, *Multi-directional search: A direct search algorithm for parallel machines*, PhD thesis, Department of Mathematical Sciences, Rice University, Houston, TX, 1989.
- [14] —, *On the convergence of pattern search methods*, SIAM Journal on Optimization, (1997). To appear.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY(Leave blank)	2. REPORT DATE March 1996	3. REPORT TYPE AND DATES COVERED Contractor Report		
4. TITLE AND SUBTITLE PATTERN SEARCH ALGORITHMS FOR BOUND CONSTRAINED MINIMIZATION		5. FUNDING NUMBERS  C NAS1-19480 WU 505-90-52-01		
6. AUTHOR(S) Robert Michael Lewis Virginia Torczon				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Institute for Computer Applications in Science and Engineering Mail Stop 132C, NASA Langley Research Center Hampton, VA 23681-0001		8. PERFORMING ORGANIZATION REPORT NUMBER  ICASE Report No. 96-20		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Langley Research Center Hampton, VA 23681-0001		10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA CR-198306 ICASE Report No. 96-20		
11. SUPPLEMENTARY NOTES Langley Technical Monitor: Dennis M. Bushnell Final Report Submitted to SIAM Journal on Optimization.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT  Unclassified-Unlimited  Subject Category 64		12b. DISTRIBUTION CODE		
13. ABSTRACT (Maximum 200 words) We present a convergence theory for pattern search methods for solving bound constrained nonlinear programs. The analysis relies on the abstract structure of pattern search methods and an understanding of how the pattern interacts with the bound constraints. This analysis makes it possible to develop pattern search methods for bound constrained problems while only slightly restricting the flexibility present in pattern search methods for unconstrained problems. We prove global convergence despite the fact that pattern search methods do not have explicit information concerning the gradient and its projection onto the feasible region and consequently are unable to enforce explicitly a notion of sufficient feasible decrease.				
14. SUBJECT TERMS bound constrained optimization; pattern search methods; direct search methods			15. NUMBER OF PAGES 21	
			16. PRICE CODE A03	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT	